



Face Feature Extraction Based on Agents With Multi-camera System

Yanfei Zhu¹, Qiuqi Ruan^{1,2}

¹Institute of Information Science, Beijing Jiaotong University, Beijing, China

²Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing, China

Email:10120416@bjtu.edu.cn

(Abstract) Face feature extraction is a complex and challenging issue in face recognition, it has important affect on recognition rate. A novel approach was proposed to extract face feature base on Agents with multi-camera system. In this approach, we use several cameras to solve perception problems caused by gesture and other complex conditions. According to the ability of cooperation and intelligence of the Agent, we can give each camera an Agent, each Agent tries to get the pose parameters of face feature from the environment. Those Agents also compete with each other, in order to become the best facial object. In this way the face feature can be better extracted with the Agent system. It can be seen from our experiment in images which collected by our system that this method can better extract face feature, which solves the problem of false recognition caused by angles, and thanks to the parallelism of the Agent platform, the system has an advantage in process time.

Key Words: Multi-Camera: Face Feature Extraction: Agent: Competition Mechanism

1. INTRODUCTION

Face recognition has always been one of the most important research contents of pattern recognition and image processing[1][2], while face feature extraction[3][4] has big affect on recognition rate. Traditional algorithms of face feature extraction are always affected by the illumination, pose, and other complicated circumstances, so are affected of the recognition rate. In recent researches, many algorithms of face feature extraction which have strong robust are proposed. Pentland[5] proposes a method which uses PCA algorithm in some important regions of the face, according to the result, we can see that because of less sensitive to the variation of pose, illumination, the locality methods are sometimes better than the global methods. While Du[6] solves this problem from face preprocessing, he proposes a logarithmic edge algorithm of illumination processing. This method decreases the affect on recognition rate of illumination. Those kinds of methods improve the feature extraction from the algorithm aspect, while in this way we can not adapt to the complicated circumstances completely. Here we create a multi camera system, and thanks to several cameras, we can solve the problems which caused by illumination, poses from the perception section. How to make these cameras work together and cooperate with each other brings a problem, Agent theory and technique offers a scheme to solve this issue. Chen[7] proposes a pattern recognition framework based on Agent, he gives a viewpoint that we can know the pattern from different sides, and due to the collaboration of each Agent we can get

the whole cognition of the pattern.

We use several Agents which has their own behaviors to agent the cameras, each Agent control one camera. These Agents get the facial gesture parameters[8] from perception side according to some given rules, and transmit their own parameters to the main agent with the communication ability of the Agent. During the transfer the main Agent call a behavior of competition which makes all the control Agents compete with each other with their gesture parameters to become the final best face picture, in the mean time we get the best face feature. It can be seen from our experiment in images which collected by our system that the Agent method can better extract face feature, which solves the problem caused by angles, and due to the parallelism processing of the Agent platform, the system also has an advantage in process time.

2. AGENT FRAMEWORK THEORY

Agent in MAS(Multi-Agent system) has a wild definition, we always consider it as an autonomy system which can achieve his own design goal by his perception of the circumstance and his ability of independent behaviors and cooperation. In this approach we use an Agent system to control the multi camera system to extract the face feature. Every Agent in the system can get the face feature by perceiving the environment, while in this process we can use some given rules to calculate the facial pose parameters. Each Agent will communicate with others with their pose parameters and cooperate with others to decide which facial pose the Agent takes is the best, then we get the face feature from this Agent. The framework of the Agent-based multi camera system is shown in **Figure 1**.

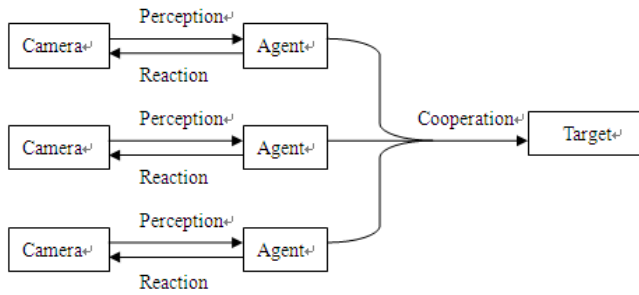


Figure 1. Agent-based multi camera system

2.1. Facial Pose Parameters

For every Agent, it will perceive the environment to get the face feature information from different angles, so the pose parameters each Agent calculates will vary from the angles. In our competition mechanism, the main Agent will get the final best pose parameters of the face from all the control Agents. In this way we define three facial pose parameters as a factor of the competition mechanism, we can put the face in a three-dimensional coordinate system. Each direction will become a parameter, we can call them Pitch, Yaw and Roll, as shown in Figure 2.

Erik and Mohan[8] make a survey of current algorithm researches of the head pose estimation, they also analyze the advantages and disadvantages of each head pose estimation system. Based on their survey we will define how to get these three pose parameters for competition.

We first extract every single face feature of each Agent using ASM [10] (Active Shape Model) to locate the characteristic point of eyes, mouse and nose of the face. We also save these points as the reserve to extract for the main Agent. Figure 3 shows the result of face feature extraction by ASM, here we use 68 mark points.

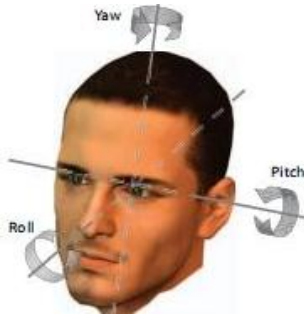


Figure 2 Three-dimensional coordinate in face

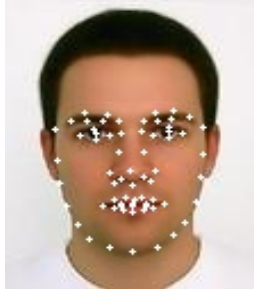


Figure 3 Position 68 features using ASM

In general, we use the direction of gaze to describe the pose of the face, which is defined as the rectilinear direction of our

eye level. Wu[11] first analyze the two express methods of the facial pose (section normal method and rotation angle method) and get the connection of these two methods. He uses the five feature positions which include two outside corners of the eyes, two corners of the mouse and the nose to calculate the two kinds of expression of the facial pose. Due to the theory that any kind pose of the face can rotate from the normal face

with α angle for Z axis, β angle for Y axis and φ angle for X axis. So if we know the direction of the gaze of the face relevant to the specific facial pose, then we rotate it with the

angle φ for X axis, β for Y axis, α for Z axis, and we can

get the normal front face. As described before, we can give the

definition of our three facial pose parameters as below:

$$\text{Pitch} = -\varphi', \text{Yaw} = -\beta', \text{Roll} = -\alpha'$$

In order to make our calculation simple, we use the Euler angles of the rotated angles for convenience, these parameters will be the factors of the competition mechanism to get the best facial pose picture.

2.2. The Competition Mechanism of Agent

In our system, each Agent will get their own face parameters, and then communicate with other Agents within these parameters. During the communication process, the main Agent will wake up a series of behaviors of the control Agent. These behaviors will try to make their Agent become the best feature target of the system. In general, we consider the positive face contain the most feature information, while other angles of the face will lose some feature information, so we define a rule for the standard of the best facial pose, that is the face whose pose have the minimum angle deflection of the positive face will be chosen to be the best facial pose.

When the competition mechanism process begins, the main Agent will provide a facial pose target for all the control Agents to compete for, only the Agent who has the best facial pose will give the target its own value. According to the definitions of the three facial pose parameters before, we can first get the angle deflection value of each coordinate of the face. The value of the facial pose target can be defined as the sum of all the three deflection absolute values. For convince, we use Euler angle for calculation. The definition of the value of the facial pose target is shown below:

$$\text{Value} = \text{abs}\left(\frac{180 \times V1}{\pi}\right) + \text{abs}\left(\frac{180 \times V2}{\pi}\right) + \text{abs}\left(\frac{180 \times V3}{\pi}\right) \quad (1)$$

Where $V1$, $V2$, $V3$ respect the Euler angles of each direction of the three-dimensional coordinate system respectively, and the value is the target facial pose value.

During the competition mechanism process, the rule of the competition is:

$$\text{Value} = \begin{cases} \text{temp}, & \text{if } (\text{temp} > \text{Value}) \\ \text{Value}, & \text{else} \end{cases} \quad (2)$$

Where temp respects the value which the Agent brings itself

3. SYSTEM IMPLEMENTATION METHOD

To implement our Agent system, we use JADE (Java Agent Development Framework) with Netbeans to achieve it. While using OpenCV library to realize the face feature extraction and facial pose estimation part. Finally we can utilize JNI standard to combine c language platform with the java language platform, in this way we can share code resources in the two platforms.

3.1. JADE Agent Operation Theory

JADE[12] is an Agent system created by java, which goal is to simplify the development process of Agent system by following the comprehensible system service and the specification of the main set. It can be used to develop an application based on Agent, which keeps to the FIPA specification, and can achieve a goal that all the Agent system can communicate with each other. The JADE Agent is performed as an entity which has the ability of autonomy, communication and cooperation.

We first create a MainAgent in the main container, which used to control and manage all the Agents in the platform. In JADE, the ability of an Agent to execute its own task is called Behavior, Agent can also scheme its behavior due to its own demand and ability. Here we define that the behavior is an abstract class, which offers a basic task framework. The flow chat of how an Agent executes its behavior is shown in **Figure 4**.

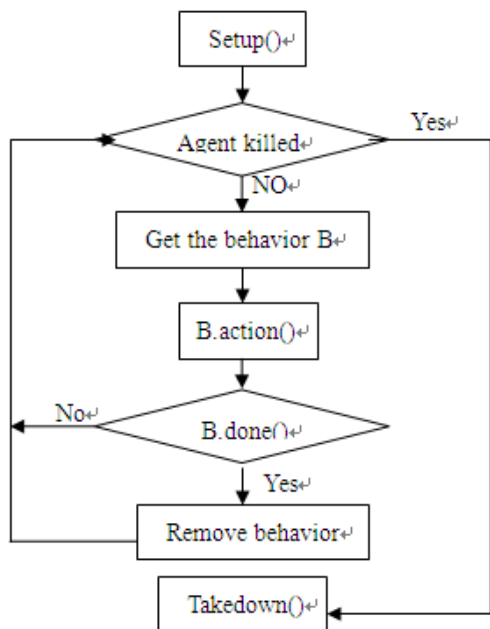


Figure 4 Flow of how an Agent execute its behavior

In our Agent framework, the action function of the MainAgent can be expressed by a triple table:

MainAgent.action ::= < Create, Control, Receive >

- Create attribute: the MainAgent will first create some ControlAgent, the number is the same as the number of cameras, and also tell the ControlAgent its objection of perception, i.e. the path of the face picture to perceive the facial environment.
- Control attribute: manage and control all the ControlAgent in the platform, in our approach, it mainly manages the competition mechanism.
- Receive attribute: when the competition is finished, the best facial pose has been chosen, in this way we send face feature of this facial pose to the MainAgent.

The ControlAgent can also expressed by a triple table:

ControlAgent.action ::= < Perception, Competition, Send >

- Perception attribute: perceive the objection environment, extract face features, and calculate the facial pose parameters based on some given rules.
- Competition attribute: the ControlAgent will communicate and exchange information with other ControlAgent, and then compete with each other for the facial pose target value with their own pose parameters, the parameters will calculate by **Eq.1**.
- Send attribute: the final result of the competition is judged by **Eq.2**, after the best facial pose is chosen, the ControlAgent will send the face feature of this pose picture to the MainAgent.

The message structure of Agent communication use ACL (Agent Communication Language) language, which can realize the communication and information exchange of the Agents conveniently. Here we use sendMessage() and receiveMessage() command to send and receive message, when the ControlAgent send the best face feature to the MainAgent, we achieve this process through a group of Java serialized objects. All the messages no matter sent or received will follow queuing mechanisms, i.e. they are sequenced to wait for message processing. When the competition of all the ControlAgent is over, in the mean time, there is no other messages will sent or received to the MainAgent, we define this moment that the face feature extraction is finished.

3.2. JNI Standard

Recent years, there are so many researches of face feature extraction, and some have been developed to an open source library. Here we use C language combining OpenCV library to achieve the functions of face feature extraction and facial pose estimation. In this way we get a problem that how can these functions created by C and OpenCV library used in the java platform, and JNI standard offers us a solution. JNI(Java Native Interface) acts like a bridge for the C and java platform, which allows java codes use other language source. The flow of how to use JNI is shown in **Figure 5**.

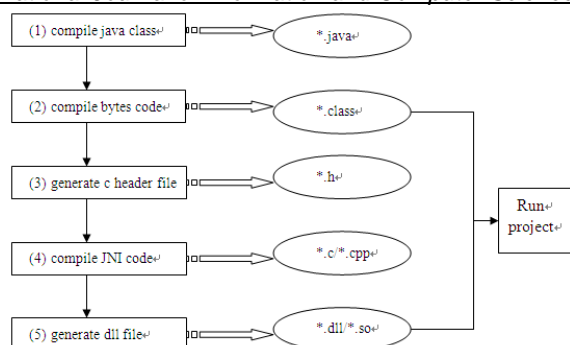


Figure 5 The flow of JNI standard

Agent in our system is created in java environment, in this way, calling c functions in java is called native function. When we construct the ControlAgent in java platform, we should first declare that the function to get the facial pose parameters is a native function. Then we get HandleFace.h by the command javah. In VS we can use OpenCV to realize the function of HandleFace(), all these files will compile to a dll project. ControlAgent will get the c functions in static with the command System.loadLibrary("HandleFace").

4. EXPERIMENT RESULTS

In order to verify our approach, we use multi camera system to collect a couple of face pictures, which are taken from different angles of three cameras. The original picture is shown in **Figure 6**.

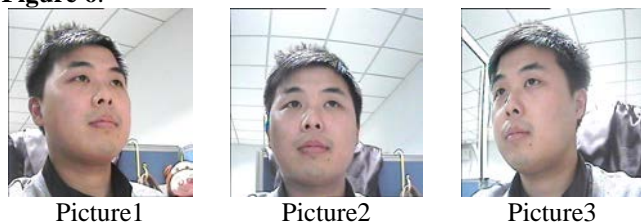


Figure 6. Original pictures

Multi camera system makes us get more face information in perception step, and then we try to get the better face feature with the information. When the MainAgent has been created, each ControlAgent will get their own face features, and calculate its facial pose parameters. The parameters of each Agent list in **Table 1**. After the competition mechanism, the MainAgent will get the picture of best facial pose (picture2, Value = 33.31478). In order to observe the facial pose directly, we draw the face region and the three-dimensional coordinate of the positive face, as shown in **Figure 7**. It is obvious that the face feature extracted from picture2 is better than picture1 and picture3, as its deflection angle is smaller. Finally the MainAgent will store the feature in picture2 for further research.

Table 1 The three direction value of each picture

ATTR IBUTE	VALUE			
Object	Pitch	Yaw	Roll	Value
Picture1	-9.76715020	18.48283350	21.072994	49.32297650
Picture2	0.484889	16.411161	16.418734	33.314784
Picture3	10.282576	16.457791	19.563668	46.304034

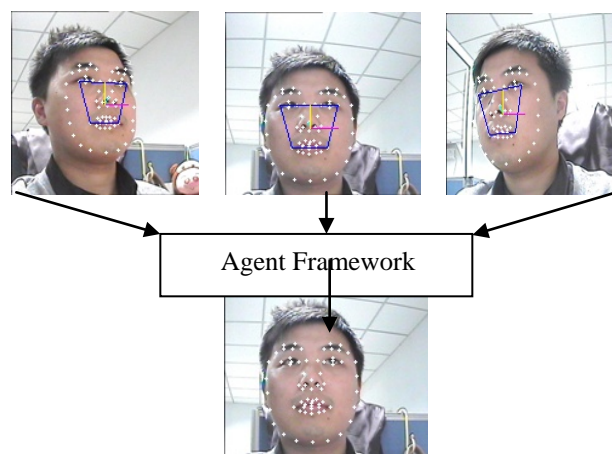


Figure 7 The Agent competition result

In order to verify the effectiveness of our approach to solve the face feature extraction problem caused by deflection angle, we use one camera system and multi camera system to collect 40 face pictures from different angles respectively. Then we use these face features to do face recognition, here we use ORL face library. the result of the recognition is shown in **Table 2**. We can see that the face chosen by our multi Agent system has a better recognition rate.

Table 2 The recognition rate of two systems

Test pictures	Recognition rate
One camera system	0.7500
Agent platform	0.8500

Thanks to the parallelism processing of the Agent platform, each Agent is an independent entity, and all Agents run and handle messages at the same time. In our system, each running Agent is an independent thread, so the processing time of the system has a big advantage. **Table 3** shows the time of each face feature extraction needs and the time in our Agent platform handling three face features needs. We can see that when we handle three faces respectively, we will need $1534+1668+1822=5024\text{ms}$ in total, while in the Agent platform we only need 1965ms to extract all three face features and chose the better one. It is obvious that Agent platform improve the processing efficiency of the system.

Table 3 The processing time in our system

Object	Picture1	Picture2	Picture3	Agent
Time(ms)	1534	1668	1822	1965

5. CONCLUSIONS

An approach of face feature extraction based on Agent using multi camera system is presented in this paper. Multi camera system is used to get more information in perception step, while Agent platform can make this system autonomy and faster. Each Agent can get the facial pose parameters by perceiving the environment, through the competition mechanism the MainAgent can get the best facial pose value, and store the features of this face. JNI standard makes the c platform and java platform can share code source with each other. The

experiment results show that this approach can solve the problems of feature extraction brings by deflection angles effectively. Due to the parallelism processing of the Agent platform, the system has a big advantage in handling time. Above all, our approach has a wild application future.

There remain some open problems that are to be addressed in our future research. For instance, it would be interesting to find out an optimal Agent framework of best extraction. There are two considerations in our design of this framework. First, the competition mechanism is just a simple way the choose the best pose of the three faces. So we want a better interaction mechanism for Agent to maximize the use of all face information to create a better face feature, for example, a data fusion mechanism. Second, we wish to see that the system is accomplished in the most efficient way.

6. ACKNOWLEDGEMENTS

This work is supported by Chinese National Natural Science Foundation (60973060) and Research Fund for Doctoral Program (200800040008).

REFERENCES

- [1] Kaya Y, Kobayashi K, A basic study on human recognition, In *Frontiers of Pattern Recognition*, New York: Academic, 1971:265-289.
- [2] P.J.Phillips, H.Moon, The FERET Evaluation Methodology for Face-Recognition Algorithms, *IEEE Transactions on PAMI*, 22(10):1090-1104(2000).
- [3] Chen C, Huang C, Human Facial Feature Extraction for face interpretation and recognition, *Patten Recognition*, 25(12):1435-1444(1992).
- [4] G.M. Beumer, Q. Tao, A.M. Bazen, R.N.J. Veldhuis, A Landmark Paper in Face Recognition, *Seventh IEEE International Conference on Automatic Face and Gesture Recognition*, 2006: 73-78.
- [5] A. Pentland, B. Moghaddam, T. Starner, View-Based and Modular Eigenspaces for Face Recognition, *IEEE CS Conf. Computer Vision and Pattern Recognition*, 1994:84-91.
- [6] Du Bo, Research on illumination Preprocessing Approaches in Face Recognition, Beijing: Institute of Computing Technology, Academia of Sciences(2005).
- [7] Chen Xianyi, A pattern recognition framework based on multi Agent, *CAAI Transactions on Intelligent Systems*, 1(2):89-93(2006).
- [8] Erik Murphy-Chutorian, Mohan Manubhai Trivedi, Head Pose Estimation in Computer Vision: A Survey, *IEEE Transactions on Patten Analysis and Machine Intelligence*, 31(4):607-626(2008).
- [9] Dale J, A mobile agent architecture for distributed information Management, Ph.D. Thesis, University of Southampton(1997).
- [10] Cootes, C.J. Taylor, Active Shape Models-Smart Snakes, *Proc. British Machine Vision Conference*, Springer-Verlag, 1992:266-276.
- [11] A. Gee, R Cipolla, Estimating gaze from a single view of face, *International Conference on Pattern recognition*, A:758-760(1994).
- [12] Fabio Bellifemine, Giovanni Caire, Dominic Greenwood, developing multi-Agent system with JADE, West Sussex: WILEY(2007).